Facultad de Informática
Grado en Ingeniería Informática

Lógica

PARTE 3: DEMOSTRACIÓN AUTOMÁTICA

Tema 11: Forma Normal de Skolem

Profesor: Javier Bajo
jbajo@fi.upm.es

Introducción a la lógica.

❑ Componentes

Parte 1
Lógica Proposicional

Parte 2
Lógica de Primer Orden

Parte 3
Demostración Automática

Parte 4
Resolución

Introducción a la demostración automática.

- El **razonamiento automático** se dedica a estudiar cómo usar un ordenador para ayudar en la parte de resolución de problemas que requiere razonamiento.
- Se trata de implementar programas que verifiquen un razonamiento mediante una serie de pasos de inferencia.
- Se suele denominar **deducción automática** porque se suele utilizar razonamiento como proceso deductivo.
- Cuando el trabajo se centra en la obtención de algoritmos que permitan encontrar pruebas de teoremas matemáticos, recibe el nombre de ***demostración automática de teoremas*** **(DAT)**.

- Algunas cuestiones que surgen durante dicho estudio son la representación del conocimiento, las reglas para derivar nuevo conocimiento del que se tiene, y las estrategias para controlar dichas reglas.
- Otras cuestiones se refieren a la implementación de la teoría resultante y a las aplicaciones para las cuales el correspondiente software puede ser usado. Teoría, implementación y aplicaciones juegan papeles vitales para el razonamiento automático a la hora de intentar alcanzar uno de sus principales objetivos: proporcionar un asistente de razonamiento automático.

Introducción a la demostración automática.

- El razonamiento automático se dedica al **desarrollo de programas de ordenador que sean capaces de demostrar que una conjetura es una consecuencia lógica de un conjunto de axiomas o hipótesis**.

  - El lenguaje en el que se escriben la conjetura, las hipótesis y los axiomas es una **lógica**, a menudo de primer orden, pero también puede ser de orden superior.
  - Las pruebas producidas por un sistema de razonamiento automático describen cómo y por qué la conjetura es una consecuencia de los axiomas y las hipótesis, utilizando para ello las reglas de derivación.

  - Los **pasos a seguir** son:
    - El usuario formaliza el problema y lo pasa al sistema de demostración automática como entrada.
    - El sistema intenta resolver el problema:
      - Si el sistema tiene éxito, se obtendrá una solución al problema.
      - Si el sistema no tiene éxito, entonces el usuario puede proporcionar cierta ayuda, intentar demostrar un resultado intermedio o revisar la formalización.

Introducción a la demostración automática.

- De forma general, podemos hablar de **dos tipos de aproximaciones** principales.

  - Atendiendo a la forma en la que se simula el razonamiento. Inicialmente se utilizó inferencia por instanciación. Actualmente se utiliza el **principio de resolución de Robinson** (lo veremos en el tema 14), que permite la obtención de conclusiones generales a partir de premisas generales (aplicando lo menos posible la regla de instanciación).

  - Atendiendo a la **cantidad de información intermedia retenida**. 3 alternativas:
    - Almacenar toda la información derivada intermedia, con el consiguiente crecimiento del espacio de búsqueda.
    - No almacenar ninguna información intermedia, con la consiguiente perdida de potencia deductiva.
    - Almacenar solo información que cumpla ciertos requisitos. Ejemplos de esta alternativa son la estrategia de resolución semántica y los métodos para el tratamiento de la información intermedia (Wos).

Introducción a la demostración automática.

- Los D.A.T.:
  - Hacen uso de una representación especial de las formulas: la **forma clausal o clausular**.
  - En la deducción se emplean reglas de inferencia alejadas del razonamiento humano, como **resolución** y para-modulación.
  - Por ultimo, destacar que en la mayoría de los casos, los demostradores automáticos de teoremas realizan **pruebas por contradicción (o refutación)**.

  - Las **acciones típicas** que lleva a cabo un demostrador de teoremas, son:
    1. Aplicación de las reglas de inferencia para obtener conclusiones.

    2. Cuando se obtiene una conclusión, el programa la rescribe a una forma canónica para ver si es un corolario trivial de información que el programa ya posee y, por consiguiente, debe desecharse.

    3. Si la conclusión obtenida entra en contradicción con alguna de las informaciones mantenidas por el demostrador, entonces la prueba concluye con éxito (recordemos que hablamos de pruebas por refutación), en caso contrario se vuelve a repetir el proceso.

- **Límites** de la demostración automática de teoremas:

  - **Límites teóricos**:
    - La lógica de primer orden es indecidible: no existe un procedimiento general que permita determinar si una formula A es o no un teorema (lógicamente valida en el lenguaje de primer orden L). Sin embargo existen métodos de semi-decisión, donde lo introducido es falso si el procedimiento no acaba.
    - Otro limite viene impuesto por la complejidad computacional de los problemas objeto de estudio y los algoritmos utilizados en su solución.

  - **Límites prácticos**.
    - Crecimiento exponencial del espacio de búsqueda (conocido como explosión combinatoria). Este problema obliga a una necesidad creciente de memoria para el almacenamiento de estructuras intermedias utilizadas en los cómputos, que sobrepasa las posibilidades de las maquinas actuales.
    - Ineficiencia: alto coste temporal de los algoritmos utilizados en el proceso de la demostración.

Introducción a la demostración automática.

- Ejemplos.

  - **Rompecabezas lógicos**:
    Tres misioneros y tres caníbales están en la orilla derecha de un río. Hay una barca que puede transportar sólo dos personas. Si el número de misioneros nunca pude ser inferior al de caníbales en ninguna de las orillas, ¿cómo pueden cruzar todos a la orilla izquierda?. (La barca no puede cruzar vacía).

  - **Diseño de circuitos**:
    Usando cualquier número de puertas AND y OR, pero no más de dos puertas NOT, construir un circuito de acuerdo con la siguiente especificación: Hay tres entradas (e1, e2 y e3) y tres salidas (s1, s2 y s3) de forma que se cumpla si =not(ei).

  - **Problemas matemáticos**:
    Sea G un grupo y e su elemento neutro. Demostrar que si, para todo x de G, x2 = e, entonces G es conmutativo.

Estandarización de fórmulas

- ¿Cómo trabajar con DAT?
- Para trabajar con DAT es necesario trabajar con **fórmulas estandarizadas**.

- Objetivo: **simplificar las fórmulas**

  Queremos obtener, mediante una serie de ***transformaciones***, una fórmula que sea más fácil de manipular automáticamente, pero que siga teniendo ciertas propiedades de la fórmula original (estandarización):

Fórmula en un lenguaje de primer orden	$\exists y\ (\forall x\ (p(x,f(y)) \rightarrow q(z)) \vee \neg\exists w\ p(g(w),y))$
⇓	⇓
Fórmula en forma normal de Skolem	$\forall x\ \forall w\ (\neg p(x,f(b)) \vee q(a) \vee \neg p(g(w),b))$
⇓	⇓
Fórmula en forma clausular	$\{\neg p(x,f(b)) \vee q(a) \vee \neg p(g(w),b)\}$

Estandarización de fórmulas

- ¿Qué es lo que preserva esta transformación?
  - preserva la **satisfacibilidad**
  - pero no preserva todos los modelos (es decir, la semántica): el resultado **no** es equivalente a la fórmula original

- Preservación

  Consideremos una transformación de F a F'
  - preservar la **semántica** significa que, para toda interpretación I, I es un modelo de F sii es un modelo de F'

    $\forall x p(x)$ es semánticamente equivalente a $\neg\exists x \neg p(x)$

  - preservar la **satisfacibilidad** significa que existe un modelo I de F sii existe un modelo I' (probablemente no el mismo) de F'

    $SAT(\exists x p(x))$ sii $SAT(p(a))$

Forma Normal de Skolem.

- **Formas Normales.** El objetivo de la estandarización de fórmulas es reducir la variedad sintáctica de un LPO, uniformando sus fórmulas.
- La idea es que la **fórmula inicial es satisfacible sii su transformada es satisfacible**.

  - Reducción de la multiplicidad de conectivas (formas normales en la lógica proposicional).
    - Forma normal conjuntiva. FNC
    - Forma normal disyuntiva. FND
  - Uniformización de la posición de los cuantificadores en las fórmulas (Formas Normales en la LPO).
    - Forma Prenex. FNP
  - Cierre de fórmulas abiertas (cuantificación existencial de variables libres).
  - Eliminación de cuantificadores existenciales.
    - Forma de Skolem. FNS
  - Forma Clausular (variante sintáctica de la Forma Skolem).

Forma Normal de Skolem.

- **Forma Normal Disyuntiva.** Una fbf A está en forma normal disyuntiva si y solo si $A \equiv A_1 \vee A_2 \vee A_3 \vee \ldots \vee A_n$ con $n \geq 1$, y cada A_i es una conjunción de literales.

  Ejemplo: $(p \wedge \neg q \wedge r) \vee \neg p \vee (r \wedge \neg s)$

- **Forma Normal Conjuntiva.** Una fbf A está en forma normal conjuntiva si y solo si $A \equiv A_1 \wedge A_2 \wedge A_3 \wedge \ldots \wedge A_n$ con $n \geq 1$, y cada A_i es una disyunción de literales.

  Ejemplo: $(q \vee p) \wedge (\neg r \vee s) \wedge \neg q$

- **Forma normal prenexa:** Una fbf A está en forma normal prenexa si y solo si $A \equiv (Q_1 X_1)(Q_2 X_2)\ldots(Q_n X_n)M$, donde $X_1, X_2 \ldots X_n$ son variables diferentes y $Q_i \equiv \exists$ o $Q_i \equiv \forall$, para cada $1 \leq i \leq n$. M es una fórmula que no contiene cuantificadores. Al componente $(Q_1 X_1)(Q_2 X_2)\ldots(Q_n X_n)$ se le llama prefijo, y a M matriz de la fórmula A.

Forma Normal de Skolem.

❑ **Forma normal prenexa.**

Ejemplos: Están las siguientes fórmulas en FNP?

- $\neg\forall x\,[P(x) \rightarrow \exists x\, P(x)]$ no
- $\exists x \forall y\,[P(x) \wedge \neg P(y)]$ si
- $\exists x\, P(x) \vee \forall y\, Q(y)$ no
- $\exists x \forall y\,[P(x) \vee Q(y)]$ si
- $\forall y \exists x\,[P(x) \vee Q(y)]$ si
- $\neg(\exists x\,[P(x) \rightarrow Q(x)] \rightarrow \exists x\,[P(x) \rightarrow R(x)])$ no
- $\forall z \exists x \exists y\,[((\neg P(x) \vee Q(x)) \wedge (\neg Q(y) \vee R(y))) \wedge P(z)]$ si

Forma Normal de Skolem.

❑ **Forma Normal de Skolem**.

- Def.: La fórmula F está en **forma de Skolem** (FS) si es de la forma $\forall x_1 \ldots \forall x_n\, G$, donde $n > 0$ y G no tiene cuantificadores.

Es decir:

- Todos los cuantificadores a la cabeza de la fórmula (forma Prenex).
- No hay variables libres.
- Sólo hay cuantificadores universales.
- Matriz de la fórmula en forma normal conjuntiva (FNC, conjunción de disyunciones de literales).

Ejemplos:

  - $\forall x \exists y P(x, y)$ no está en forma de Skolem
  - $\forall x P(x, f(x))$ sí está en forma de Skolem
  - $\exists x Q(x)$ no está en forma de Skolem
  - $Q(a)$ sí está en forma de Skolem

Forma Normal de Skolem.

- **Procedimiento para hallar FNS(A)**:

1. Dada una fórmula *A*, poner la fórmula en forma Prenex.

   Prenex (A)

2. Realizar el cierre existencial.

   Cierre(Prenex(A)) = QM, donde *Q* y *M* son el prefijo y la matriz cuantificacionales de la fórmula, respectivamente.

3. Poner la fórmula en forma normal conjuntiva.

   Obtención de *FNC(M)*

4. Eliminar los cuantificadores existenciales.

   Obtención de *Skolem(QFNC(M)) = FNS(A)*

Forma Normal de Skolem.

1. **Forma normal prenexa.** Aplicando a una fórmula los siguientes pasos se obtiene otra fórmula equivalente y que está en forma normal prenexa:

   - Rectificar la fórmula usando las equivalencias. Cambio de nombre de variables ligadas:

     $$\vdash \forall x A(x) \leftrightarrow \forall y A(x/y)$$
     $$\vdash \exists x A(x) \leftrightarrow \exists y A(x/y)$$

   donde y es una variable que no ocurre libre en A.

   - Interdefinición de cuantificadores (interiorizar negaciones usando equivalencias).

     $$\vdash \neg\forall x A(x) \leftrightarrow \exists x \neg A(x)$$
     $$\vdash \neg\exists x A(x) \leftrightarrow \forall x \neg A(x)$$

Forma Normal de Skolem.

- 3. Distribución de conectivas respecto a cuantificadores.

$$\vdash \forall xA \wedge C \leftrightarrow \forall x(A \wedge C)$$
$$\vdash \exists xA \wedge C \leftrightarrow \exists x(A \wedge C)$$

$$\vdash (\forall xA \rightarrow C) \leftrightarrow \exists x(A \rightarrow C)$$
$$\vdash (A \rightarrow \forall xC) \leftrightarrow \forall x(A \rightarrow C)$$
$$\vdash (\exists xA \rightarrow C) \leftrightarrow \forall x(A \rightarrow C)$$
$$\vdash (A \rightarrow \exists xC) \leftrightarrow \exists x(A \rightarrow C)$$

$$\vdash \forall xA \vee C \leftrightarrow \forall x(A \vee C)$$
$$\vdash \exists xA \vee C \leftrightarrow \exists x(A \vee C)$$

si x no está libre en la otra subfórmula

$$\vdash \forall xA \wedge \forall xC \leftrightarrow \forall x(A \wedge C)$$
$$\vdash (\exists xA \vee \exists xC) \leftrightarrow \exists x(A \vee C)$$

Lema: Para toda fórmula A, $\vdash A \leftrightarrow Prenex(A)$

Lema: La forma Prenex de una fórmula siempre existe, aunque puede no ser única

Forma Normal de Skolem.

2. **Cierre Existencial.** Las variables libres de la fórmula se ligan existencialmente poniendo el cuantificador correspondiente en cabeza de la fórmula:

Lema: Una fórmula A(x) es satisfacible sii $\exists x A(x)$ es satisfacible

- Ejemplo:

$$\forall y \exists z (p(x) \wedge q(y) \rightarrow r(f(z), x))$$

se transformaría en

$$\exists x \forall y \exists z (p(x) \wedge q(y) \rightarrow r(f(z), x))$$

Forma Normal de Skolem.

3. **Forma Normal Conjuntiva - FNC.** Se utilizarán los siguientes teoremas de equivalencia:

- Interdefinición (Eliminar bicondicionales y condicionales usando la equivalencia):

$$\vdash (A \rightarrow B) \leftrightarrow (\neg A \vee B)$$
$$\vdash (A \leftrightarrow B) \leftrightarrow (A \rightarrow B) \wedge (B \rightarrow A)$$

- Leyes de De Morgan (interiorizar negaciones usando equivalencias):

$$\vdash \neg(A \wedge B) \leftrightarrow \neg A \vee \neg B$$
$$\vdash \neg(A \vee B) \leftrightarrow \neg A \wedge \neg B$$

- Distribución de $\vee$ y $\wedge$:

$$\vdash A \wedge (B \vee C) \leftrightarrow (A \wedge B) \vee (A \wedge C)$$
$$\vdash A \vee (B \wedge C) \leftrightarrow (A \vee B) \wedge (A \vee C)$$

Lema: Para toda fórmula A, $\vdash A \leftrightarrow FNC(A)$
Lema: La forma normal conjuntiva de una fórmula siempre existe

Forma Normal de Skolem.

4. **Eliminación de cuantificadores existenciales.** Se elimina el cuantificador existencial sustituyendo la variable que ligaba por una función de Skolem o constante de Skolem.

- La función de Skolem será:
  - Una función nueva en la fórmula, aplicada a todas la variables cuantificadas universalmente que aparecen antes que el cuantificador existencial a eliminar.
  - Si no hay tales variables se utilizará una constante nueva en la fórmula para hacer la sustitución.
- Ejemplos:
  - $\forall x \exists y(p(x) \rightarrow \neg q(y))$ se transformaría en $\forall x(p(x) \rightarrow \neg q(f(x)))$
  - $\exists x \forall z(q(x,z) \vee r(a,x))$ se transformaría en $\forall z(q(b,z) \vee r(a,b))$
  - $\exists x \forall y \exists z(p(x) \wedge q(y) \rightarrow r(f(z), x))$ se transformaría en $\forall y(p(a) \wedge q(y) \rightarrow r(f(g(y)), a))$

Lema: Una fórmula A es satisfacible sii Skolem(A) es satisfacible.

Forma Normal de Skolem.

- **Ejemplos de cálculo de FNS:**

<u>Ejemplo1</u>:

- F : $\exists x \forall y \forall z \exists u \forall v \exists w[P(x, y, z) \wedge Q(u, v) \wedge \neg R(w)]$

1.- La fórmula está en FNP.
2.- No hay variables libres.
3.- La fórmula está en FNC.
4.- Eliminar cuantificadores existenciales.

$\exists x$ no se encuentra precedido de cuantificadores universales, se sustituye por una constante *a*

$$\forall y \forall z \exists u \forall v \exists w[P(a, y, z) \wedge Q(u, v) \wedge \neg R(w)]$$

$\exists u$ y $\exists w$ están precedidos de cuantificadores universales. Serán sustituidos por fórmulas $f(y,z)$ y $g(y,z,v)$

$$\forall y \forall z \forall v\, [P(a, y, z) \wedge Q(f(y,z), v) \wedge \neg R(g(y,z,v))]$$

Forma Normal de Skolem.

Ejemplo2:

- $Sko(\forall x\ \exists y\ \forall z\ \exists w\ [\neg P(a,w) \vee Q(f\ (x), y)])$

  $= Sko(\forall x\ \forall z\ \exists w\ [\neg P(a,w) \vee Q(f\ (x), h(x))])$

  $= Sko(\forall x\ \forall z\ [\neg P(a, g(x, z)) \vee Q(f\ (x), h(x))])$

  $= \forall x\ \forall z\ [\neg P(a, g(x, z)) \vee Q(f\ (x), h(x))]$

y se sustituye por h(x)

w se sustituye por g(x,z)

Ejemplo 3:

- $Sko(\forall x\ \exists y\ \exists z\ [(\neg P(x,y) \wedge Q(x,z)) \vee R(x,y,z)])$

  Transformamos a FNC

  $\vdash A \vee (B \wedge C) \leftrightarrow (A \vee B) \wedge (A \vee C)$

  $= Sko(\forall x\ \exists y\ \exists z\ [(\neg P(x,y) \vee R(x,y,z)) \wedge (Q(x,z)) \vee R(x,y,z))])$

  Se eliminan los cuantificadores existencias

  y se sustituye por f(x)
  x se sustituye por g(x)

  $= \forall x\ [(\neg P(x,f(x)) \vee R(x, f(x), g(x))) \wedge (Q(x, g(x))) \vee R(x, f(x), g(x)))]$

Forma Normal de Skolem.

- <u>Ejemplo 4</u>: $\forall x \forall y[\exists z P(x, y, z) \wedge (\exists u Q(x, u) \rightarrow \exists v Q(y, v))]$

1. Forma prenexa: (Cambio de nombre, interiorizar negaciones, *extracción de cuantificadores*)

$\vdash \exists x A \wedge C \leftrightarrow \exists x(A \wedge C)$

- $\forall x \forall y[\exists z(P(x, y, z) \wedge (\exists u Q(x, u) \rightarrow \exists v Q(y, v)))]$
- $\forall x \forall y \exists z[P(x, y, z) \wedge (\exists u Q(x, u) \rightarrow \exists v Q(y, v))]$

$\vdash (\exists x A \rightarrow C) \leftrightarrow \forall x(A \rightarrow C)$

- $\forall x \forall y \exists z[P(x, y, z) \wedge \forall u(Q(x, u) \rightarrow \exists v Q(y, v))]$

$\vdash \forall x A \wedge C \leftrightarrow \forall x(A \wedge C)$

- $\forall x \forall y \exists z[\forall u(P(x, y, z) \wedge (Q(x, u) \rightarrow \exists v Q(y, v)))]$
- $\forall x \forall y \exists z \forall u[P(x, y, z) \wedge (Q(x, u) \rightarrow \exists v Q(y, v))]$

$\vdash (A \rightarrow \exists x C) \leftrightarrow \exists x(A \rightarrow C)$

- $\forall x \forall y \exists z \forall u[P(x, y, z) \wedge \exists v(Q(x, u) \rightarrow Q(y, v))]$

$\vdash \exists x A \wedge C \leftrightarrow \exists x(A \wedge C)$

- $\forall x \forall y \exists z \forall u[\exists v\,(P(x, y, z) \wedge (Q(x, u) \rightarrow Q(y, v)))]$
- $\forall x \forall y \exists z \forall u \exists v[P(x, y, z) \wedge (Q(x, u) \rightarrow Q(y, v))]$

Forma Normal de Skolem.

- Ejemplo 4: $\forall x \forall y[\exists z P(x, y, z) \wedge (\exists u Q(x, u) \rightarrow \exists v Q(y, v))]$

2. Cierre existencial: No hay variables libres, ya tenemos el cierre:
   - $\forall x \forall y \exists z \forall u \exists v[P(x, y, z) \wedge (Q(x, u) \rightarrow Q(y, v))]$
3. FNC (eliminar $\leftrightarrow$ y $\rightarrow$, interiorizar $\neg$ y distribuir $\vee$ y $\wedge$)
   - $\forall x \forall y \exists z \forall u \exists v[P(x, y, z) \wedge (\neg Q(x, u) \vee Q(y, v))]$

$\vdash (A \rightarrow B) \leftrightarrow (\neg A \vee B)$

4. Eliminación de cuantificadores existenciales:

z se sustituye por f(x,y)
v se sustituye por g(x,y,u)

   - $\forall x \forall y \exists z \forall u \exists v[P(x, y, f(x,y)) \wedge (\neg Q(x, u) \vee Q(y, g(x,y,u)))]$

Forma Normal de Skolem.

- Ejercicio: Obtener la FNS de la siguiente fórmula.

$$\neg\exists x\forall y(\neg C(x) \vee (B(a) \wedge D(x, y))) \wedge \neg\exists x A(x)$$

- Ejercicio: Obtener la FNS de la siguiente fórmula.

$$\exists x(\exists y A(x, y) \rightarrow \neg\forall z(B(x, z) \wedge C(z)))$$

Forma Normal de Skolem.

- **Teorema:** **Una fórmula F es satisfacible sii FNS(F) es satisfacible**.

1. F satisfacible sii Prenex(F) satisfacible (por lemas 1 y 5)
2. Prenex(F) satisfacible sii Cierre(Prenex(F)) satisfacible (por lema 2)
3. Sea Cierre(Prenex(F)) satisfacible de la forma Q.M, donde Q son los cuantificadores y M es la matriz de la fórmula. Entonces M satisfacible sii FNC(M) satisfacible (por lemas 3 y 5).
4. Pero también Q.M satisfacible sii Q.FNC(M) satisfacible, puesto que cuantifican las mismas variables en idéntica forma
5. Q.FNC(M) satisfacible sii Skolem(Q.FNC(M)) satisfacible (por lema 4). Pero Skolem(Q.FNC(M)) es precisamente FNS(F)
6. F satisfacible sii FNS(F) satisfacible (por silogismo 1, 2, 3, 4, 5)

Lema 1: Para toda fórmula A, $\models A \leftrightarrow Prenex(A)$
Lema 2: Una fórmula A(x) es satisfacible sii $\exists x A(x)$ es satisfacible
Lema 3: Para toda fórmula A, $\models A \leftrightarrow FNC(A)$
Lema 4: Una fórmula A es satisfacible sii Skolem(A) es satisfacible.
Lema 5: Si $\models F \leftrightarrow F'$ entonces: F satisfacible sii F' satisfacible

Forma Clausular.

- Sabemos que "Una fórmula F es satisfacible sii FNS(F) es satisfacible" y que "FNS(F) existe siempre para cualquier fórmula F", luego **podemos trabajar exclusivamente con fórmulas en forma normal de Skolem**.
- Para trabajar más cómodamente con fórmulas en FNS utilizaremos la forma clausular
- La mayoría de procedimientos de prueba operan por refutación. Estas pruebas se aplican a formulas en una forma denominada ***forma clausal***.

- **Cláusula:** Una cláusula es una disyunción finita de cero o más literales. Cuando la cláusula está compuesta de un solo literal diremos que es una cláusula unitaria.

$$C = L_1 \vee \ldots \vee L_n, \qquad L_i (1 \leq i \leq n) \text{ literal}$$

- La **forma clausular de una fórmula F, FC(F)**, es el conjunto de cláusulas de la FNS(F).
- La **forma clausular** se entiende como la conjunción de las cláusulas, cuyas variables están todas ellas ligadas universalmente.

Forma Clausular.

- Una vez que se tiene la Forma Normal de Skolem, se eliminan los cuantificadores universales y se pasa a FNC (como si estuviésemos trabajando en lógica proposicional).

- Finalmente se escribe como conjuntos.

- Ejemplo 1:

$$\forall x(G(H,H) \wedge (\neg F(x) \vee G(x,U(x)) \vee \neg G(H, x)))$$

Se eliminan los ∀ y se pasa a FNC (en este caso ya está en FNC):

$$G(H,H) \wedge (\neg F(x) \vee G(x,U(x)) \vee \neg G(H, x))$$

luego se escribe como conjuntos:

$$\{\{G(H,H)\}, \{\neg F(x),G(x,U(x)),\neg G(H, x)\}\}$$

Forma Clausular.

Ejemplo 2: $\forall x \exists y(A(x,y) \rightarrow B(x) \wedge C(y))$

Eliminación de la implicación y propiedad distributiva

- $\forall x \exists y(A(x,y) \rightarrow B(x) \wedge C(y)) \leftrightarrow \forall x \exists y(\neg A(x,y) \vee (B(x) \wedge C(y)))$
  $\leftrightarrow \forall x \exists y((\neg A(x,y) \vee B(x)) \wedge (\neg A(x,y) \vee C(y)))$

Forma Normal de Skolem

- $FNS(\forall x \exists y((\neg A(x,y) \vee B(x)) \wedge (\neg A(x,y) \vee C(y))))$

  $= \forall x((\neg A(x,f(x)) \vee B(x)) \wedge (\neg A(x,f(x)) \vee C(f(x))))$

Forma Clausular

- $\{ \{\neg A(x,f(x)) \vee B(x)\}, \{\neg A(x,f(x) \vee C(f(x)) \}\}$

Forma Clausular.

- Ejemplo 3:
  - *A:* $\forall x \forall y(p(x) \wedge (q(y) \vee r(a,x)))$

    FC(A): $\{\{p(x)\}, \{q(y) \vee r(a,x)\}\}$

Teorema: Una fórmula F es satisfacible sii FC(F) es satisfacible

Forma Clausular.

- **Forma Clausular de una deducción.**
- Una deducción [P1, P2, ..., Pn] |— C es correcta sii P1 $\wedge$ P2 $\wedge$ ... $\wedge$ Pn $\wedge$ $\neg$C es insatisfacible

Dada una deducción: [P1, P2, ..., Pn] |— C

1)Obtener la forma clausular de cada Pi, $1 \leq i \leq n$

2)Obtener la forma clausular de $\neg$C

3)Realizar la unión de todos los conjuntos de cláusulas

4)Comprobar la satisfacibilidad

Una deducción [P1, P2, ..., Pn] |— C es correcta sii

FC(P1 $\wedge$ P2 $\wedge$ ... $\wedge$ Pn $\wedge$ $\neg$C) es insatisfacible